Designing Syllable Models for an HMM Based Speech Recognition System

Kseniya Proença[1(✉)], Kris Demuynck[2], and Dirk Van Compernolle[1]

[1] ESAT - PSI, KULeuven, Kasteelpark Arenberg 10, 2441, 3001 Leuven, Belgium
{kseniya.proenca,dirk.vancompernolle}@esat.kuleuven.be
[2] ELIS, UGent, Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium
Kris.Demuynck@elis.ugent.be

Abstract. In this paper we present novel ways of incorporating syllable information into an HMM based speech recognition system. Syllable based acoustic modelling is appealing as syllables have certain acoustic-phonetic dependencies that can not be modeled in a pure phone based system. On the other hand, syllable based systems suffer from sparsity issues. In this paper we investigate the potential of different acoustic units such as phone, phone clusters, phones-in-syllables, demi-syllables and syllables in combination with a variety of back-off schemes. Experimental results are presented on the Wall Street Journal database. When working with traditional frame based features only, results only show minor improvements. However, we expect that the developed system will show its full potential when incorporating additional segmental features at the syllable level.

Keywords: Hidden Markov Models · Syllables · Continuous speech recognition · Back-off schemes

1 Introduction

Hidden Markov Models (HMM) are a very powerful statistical method of characterizing the observed data samples of discrete-time series [7]. HMM are a standard approach used in speech recognition for acoustic modeling. In recent years, great progress has been made in the modeling of the frame observation probabilities by the introduction of deep neural nets (DNNs). While improving on accuracy, thanks to the powerful discriminative training of the DNNs, the HMM-DNN framework still suffers from some of the intrinsic simplifications and limitations in the HMM architecture. In particular, the traditional approach to use beads-on-a-string to model states in (context-dependent)-phones is known to be overly simple for many reasons: there is the HMM frame-by-frame independence assumption; it does not allow to incorporate segmental information; and mismatches between canonical phonetic transcriptions and the observed acoustics are common. In this paper we focus on the latter two problems.

A. Ronzhin et al. (Eds.): SPECOM 2016, LNAI 9811, pp. 216–223, 2016.
DOI: 10.1007/978-3-319-43958-7_25

A syllable is an attractive unit for usage in speech technology applications for several reasons: it is much more salient than the shorter phone-unit, co-articulation tends to be stronger within than across syllables, and it is the shortest unit that contains all acoustic attributes relating to phonetics, rhythm and prosody. All syllabifications used in this paper are derived from our syllabification algorithm proposed in [11]. While using syllables as basic units is not novel, we believe that there is still room for improvement on how such syllabic information can be used best in an HMM based system.

The rest of the paper is organized as follows. We first talk about similar research. Then we describe the problem of the unit selection. Next, phone syllable-based labeling system is explained. In the end we give conclusions, discussions and further work.

2 Related Work

Decision trees, HMM, neural networks and trajectory models have all been used for speech recognition where syllable information is incorporated in the system.

Liao et al. [9] examine the use of the word or syllable context as a feature in the decision tree. This way, they introduce word- and syllable-specific models into the recognition system. Since they employ finite state transducer based ASR the syllable information is incorporated as features on the arcs in the transducer.

Jones and others [8] analyzed a phonetically annotated telephony database at the syllable level and built a set of syllable-based HMMs. Recognition performance was improved with syllable-level bigram probabilities and both word- and syllable-level insertion penalties. They built prototype HMMs for each syllable with the number of states set proportional to the number of phonemes in the syllable.

Zhang and Edmondson showed how a model of syllable articulation can be used with Pseudo-Articulatory Representations (PARs) to provide a general articulatory transcription of speech without phonetic labeling [14]. First, they establish the mapping between PARs and acoustic parameters. After that they perform recognition in three steps. The first step is the transition from the acoustic representation of the incoming signal to the PAR with feature trajectories available as a function of time. The second step makes a move from the PARs to the syllable structures and produces a sequence of the recovered syllables. The third stage focuses on the transition from the syllable patterns to the phonetic level and produces a sequence of phone labels.

Hu and colleagues proposed a recognition strategy which uses syllable-like units as the basic unit for recognition [6]. They define the criterion of grouping phonemes into syllable-like units as follows: phoneme sequences for which the boundary is difficult to detect are grouped together forming a new set of base recognition units. After syllable-like units are defined according to the set of predefined rules, word pronunciation models are generated using these units. Statistical trajectory models [4] are computed for each defined unit. Artificial neural networks or Gaussian mixture models are then trained to estimate probabilities of the units. The search is implemented using the Viterbi algorithm in a time-asynchronous manner.

Hauenstein [5] applied a hybrid Hidden Markov Model - Artificial Neural Network (HMM-ANN) recognition system to small and medium vocabulary recognition tasks using syllables as basic modelling units. Features are kept the same as in the phoneme-based baseline.

Syllable-level acoustic units were also used in [3] for large vocabulary continuous speech recognition (LVCSR) on telephone-bandwidth speech. The major innovation of their syllable system is the smooth integration of a large inventory of syllable models and a mixture of acoustic models ranging from monosyllabic words to CD phones.

3 Unit Selection

The main modeling unit in our system is a syllable. To make an accurate statistical model, it is essential to have enough data. The original phone-based recognition system had 43 context independent phones for which the problem of sparsity does not exist. The situation with syllables is different: there exist quite a lot of rare syllables. Therefore a back-off mechanism to smaller units is required.

The first back-off option we investigate are demi-syllables: a syllable-initial consonant cluster plus the first half of the vowel or a second half of the vowel and syllable-final consonant cluster [12]. If there is still not enough data to model a demi-syllable, we back-off to the phone sequence. For example, the word "`string`" is transcribed as "`strIN`". Demi-syllable back-off looks like "⟨`strI=`⟩⟨`=IN`⟩".

There are two main problems with this approach. The first one is sparsity. The second is how to divide the vowel in the middle and how to model it. To overcome this issue we propose another back-off mechanism. Instead of using demi-syllables, we model the syllable by three parts namely onset, vowel and coda in which onset and coda are optional consonant clusters. We call this the "cvc-scheme". The same example for "`string`" in cvc-version is "⟨`str.`⟩⟨`.I.`⟩⟨`.N`⟩".

The other question is how to decide, how many examples are needed to train a unit, and when to back-off. In this research we set this threshold to a 100 examples. The statistics were counted on the WSJ database training data [10]. We will report statistics for cvc-scheme in two ways: measured on the syllable lexicon independently of the number of occurrences and measured on running text. The database contains 5648 unique syllables. 78 % of the syllables do not occur more than 100 times and hence need to use the back-off scheme (going to cvc). On the other hand, if we take into consideration the syllable frequency, the back-off from syllable to cvc needs to be done only in 7.5 % of the cases. The same happens with the back-off from cvc to phones. On running text this happens in less than 0.5 % of times. Based on these results only the CVC backoff scheme will be considered in the remainder of this paper.

4 HMM with Syllables

For all our experiments we use the WSJ database [10], the CMU dictionary [13] and the SPRAAK toolkit [2]. SPRAAK (Speech Processing, Recognition and Automatic Annotation Kit) is an open source speech recognition package. It is an efficient and flexible tool that combines many of the recent advancements in automatic speech recognition with a very efficient decoder in a proven HMM architecture. Our speech recognition system consists of a preprocessing unit, the acoustic models and the language model, a lexicon and a search-engine. Preprocessing for all systems is the same consisting of filterbank features with vocal track length normalization and mida transformation [1]. Acoustic models are made for phones, onsets, codas, vowels and syllables separately. The lexicon includes phone, cvc and syllable descriptions of words. A semi-continuous HMM-GMM with a common pool of gaussians for all states is used. Decoding is done using Viterbi alignments.

4.1 CI Syllable Units

To create the initial context-independent (ci) syllable HMM model, we start from an existing cd-phone HMM system. Based on the phone segmentation we create ci-syllables. After the first iteration of training we regenerate segmentation of the training corpus and retrain the system. The number of states per unit depends on its length: 3 states for phones and between 3 and 19 states for the various syllables. We create three sub-models: syllables, consonant clusters+vowels, phones. These sub-models are independent, except for the shared Gaussian set.

4.2 CD Syllable Units

Modeling context-dependent (cd) syllables can be done in several ways. In our research we started from the ci-syllable system and use a phone-based context i.e. context-dependency is determined by the first or last phone of the neighbouring right/left syllable. We split all syllables, cvc and phones into two groups of long and short units. Units of 3 states are considered short units (su). Units having more than three states are called long units (lu). All states of the short units are context-dependent. In long units, only the first and last states are context-dependent. In the other words, a long unit is split as follows:

`[lu]`	→	`[lu]:L [lu]:C [lu]:R`
`[lu]:L, [lu]:R`		left/right context-dependent states
`[lu]:C`		the remaining 2 to 17 context-independent states in the syllable model.

4.3 Results

For our work we used the WSJ-based speaker independent acoustic training data [10]. We report word error rates (ERR, %) for the nov92 bigram 5 K closed-vocabulary test set (b05) and for the trigram 20 K open-vocabulary test set (t20). The results of our initial experiments are presented in Table 1. While creating the ci-syllable system, we trained it twice (as described above). The first line, first column shows the results after the first iteration. The second iteration was based on the retrained syllable system and improved the recognition result as shown in the second line, first column of the table.

We made extra analysis to find the problems of the recognition. Firstly, we evaluated the sub-models (syllable/cvc/phone) in two different ways: separately and in parallel and also investigated which unit was used (syllable/consonant cluster/phone) more often. Separate evaluation of the sub-models of the system means that the we limit the used units by the sub-model (last three columns). That means that if cvc sub-model is evaluated, we don't present any full-syllable information. The back-off to phones in the canonical transcription is used only with the limited amount of training data. Parallel evaluations means that all the units are presented in the canonical transcription and the system can choose which unit to use (first and second columns).

Secondly, we improved the Gaussians initialization. It is possible to initialize Gaussians from the phone model only and to share Gaussians only within the same model. This improved the result on 0.5 %. The results are shown in the third row of the table. cd-results are presented in the last line of the table.

Table 1. Results for the HMM system, WSJ, nov92, b05, phone ci-result: 6.86

	syl/cvc/ph	cvc/ph	syl	cvc	ph
Syllable system, 1st iteration	6.09	7.1	6.67	7.08	8.16
Syllable system, 2nd iteration	**5.66**	6.46	7.01	5.58	6.87
Gaussians initialization from phone model	6.63	7.6	9.42	7.58	6.95
cd-syllable system	4.15				

There are still a number of uncertainties and difficulties to solve. The main one is the sparsity issue: for long units (such as syllables and some consonant cluster) we don't have enough training data. To solve this, we need to have another back-off mechanism. Though, developing such a mechanism is not a trivial task. Another uncertainty concerns the system initialization and Gaussians distribution among units of different size. Pronunciation ambiguity also causes some problems. Depending on the phonetic writing, there might be several syllabification versions.

5 Phones with Syllabic Labels

In this approach we use phones as units, but all phones get a label indicating it's position in a syllable or/and word. This helps to solve the sparsity issues that we faced in the previous approach. Label is always added after the phone: <phone>:<label>. After that we train the system as it was done with regular phones. We worked out several labeling schemes. The first one (called **SylPosit**) is a simple indication the position of a phone in a syllable.

We use 4 labels :I, :C, :F, :S to indicate initial, central, final or single position of the phone in a syllable. For 2-phone syllables, the central label is not used.

The rest of the labeling schemes have the same idea and are explained in Table 2.

Table 2. Different schemes of syllabic labels

WordPosit	
Position in a word; Initial Central Final	I:I f:C I!:C S:C @:C n:C s:C i:F
PWPosit	
Both word and syllable positions; ":CF": a center phone in a word and final in a syllable. We use 8 labels (3 for words and 4 for syllables)	I:IS f:CI I!:CF S:CI @:CC n:CF s:CI i:FF
SylPositBound	
Mark only a syllable boundary; :I(initial), :C(central), :F(final) for position in a syllable. First phone can be marked as :F is it is an only phone in a syllable	I:F f:I I!:F S:I @:C n:F s:I i:C
SylPositCC	
Vowels :I(initial), :M(middle), :F(final), :S(single); and consonants :O(onset), :C(coda) in a syllable	I:S f:O I!:F S:O @:M n:C s:O i:F
SylPositVC	
Position-independent vowel (:V); position dependent consonant (initial, center, final)	INCIDENTS I!n-s@-d@nts
	I!:V n:f s:i @:V d:i @:V n:C t:C s:f

6 Modelling Syllable Boundary (SylBound)

In this approach we again use phones as basic units, but now we add syllable boundary ([:S:]) in the phonetic transcriptions. No observations or state are associated with it. For example:

```
PREDICTION [pri[:S:]dI!k[:S:]S@n]
```

By adding an extra syllable boundary marker and by using tri-phones only (looking one phone or syllable boundary marker to the left/right), the obtained phones models are fully context-dependent within a syllable only. Phones in a syllable initial or syllable final position are conditioned only on the presence of a syllable boundary and no longer on the specific left/right phone.

7 Experiments and Results

Experiments were carried on the same data as in previous research (WSJ). CI results are not presented as they are the same in all the labeling schemes. The starting point is the cd-phone HMM system. The results are presented in the Table 3.

Table 3. Results for WSJ experiments with phone labels.

	phon	WordPosit	SylPosit	PWPosit	SylPositCC	SylPositVC	SylPositBound	SylBound
b05	3.92	3.36	3.66	3.53	3.59	3.75	3.42	5.06
t20	7.6	7.27	7.66	7.57	7.50	7.64	7.74	8.24

The results indicate that loosing (or reducing) context-dependency information at syllable boundaries is not a good idea. That means that it is important to retain the phone-boundary dependency at syllable boundaries. The best results was shown with "word position" labeling system though all of the systems gave very similar results between each other and original phone system.

8 Discussion and Conclusions

In this research we tried two approaches for speech recognition using syllables. The first one is modelling syllables and making the back-off to onset-vowel-coda structure. The second one is done with labeling phones depending on their position in a syllable/word. We also tried the approach with inserting a syllable boundary that gave very poor results. We showed that the HMM with syllable position dependent phones gives better accuracy result than modelling complete syllable. The observed difference in results may be connected with the lack of training data or unefficient gaussians initialization and estimation.

Our work is similar to the research in [9] from Google though there are a few relevant some differences. While Liao and others take extra information about the syllable boundary, we model the syllable in more detail. For example, in the labeling system we model separately phones depending on the syllable position.

This system is being used as starting point for an exemplar-based system with syllable information. It is a done because we get consistent results with a well-established approach. This research is still carried on.

References

1. Demuynck, K., Duchateau, J., Compernolle, D.V.: Optimal feature sub-space selection based on discriminant analysis. In: Sixth European Conference on Speech Communication and Technology, EUROSPEECH 1999, Budapest, Hungary, 5–9 September 1999
2. Demuynck, K., Roelens, J., Compernolle, D.V., Wambacq, P.: Spraak: an open source "speech recognition and automatic annotation kit". In: INTERSPEECH, p. 495 (2008)
3. Ganapathiraju, A., Hamaker, J., Picone, J., Ordowski, M., Doddington, G.R.: Syllable-based large vocabulary continuous speech recognition. IEEE Trans. Speech Audio Process. **9**(4), 358–366 (2001)
4. Goldenthal, W.D.: Statistical trajectory models for phonetic recognition. Ph.D. thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics (1994)
5. Hauenstein, A.: Using syllables in a hybrid HMM-ANN recognition system. In: EUROSPEECH (1997)
6. Hu, Z., Schalkwyk, J., Barnard, E., Cole, R.A.: Speech recognition using syllable-like units. In: ICSLP (1996)
7. Huang, X., Acero, A., Hon, H.W.: Spoken Language Processing: A Guide to Theory, Algorithm and System Development. Prentice Hall PTR, Prentice-Hall, Inc., Upper Saddle River (2001)
8. Jones, R.J., Downey, S., Mason, J.S.: Continuous speech recognition using syllables. In: EUROSPEECH (1997)
9. Liao, H., Alberti, C., Bacchiani, M., Siohan, O.: Decision tree state clustering with word and syllable features. In: INTERSPEECH, pp. 2958–2961 (2010)
10. Paul, D.B., Baker, J.M.: The design for the wall street journal-based CSR corpus. In: ICSLP (1992)
11. Rogova, K., Demuynck, K., Van Compernolle, D.: Automatic syllabification using segmental conditional random fields. Comput. Linguist. Neth. J. **3**, 34–48 (2013)
12. Syrdal, A., Bennett, R., Greenspan, S.: Applied Speech Technology. Taylor & Francis, Oxford (1994). http://books.google.be/books?id=kyJBjxw3ducC
13. Carnegie Mellon Universit: CMU pronouncing dictionary (2008). http://svn.code.sf.net/p/cmusphinx/code/trunk/cmudict
14. Zhang, L., Edmondson, W.H.: Speech recognition using syllable patterns. In: INTERSPEECH (2002)